

Wykonał: Jakub Nowakowski

1. Cele sprawozdania

Sprawozdanie zostało wykonane w celu porównania działania różnych wyznaczników w celu klasyfikacji punktu względem prostej i określenie przyczyny ewentualnych różnic w działaniu.

2. Konfiguracja stanowiska

Procesor: Intel Core i7-8750H

Wersja Python'a: Python 3.7.4

3. Metodyka

Do określania położenia punktu względem prostej zostały użyte macierze 2x2 i 3x3, a do obliczania ich wyznaczników zostały użyte funkcje biblioteczne pochodzące z biblioteki numPy, oraz dla porównania funkcje zaimplementowane samodzielnie. Porównanie tych metod odbywało się na różnych zbiorach punktów, przy zastosowaniu różnych zakresów precyzji.

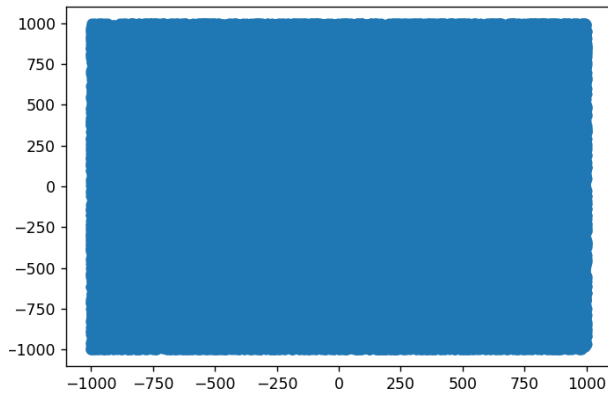
4. Zbiory, na których został przeprowadzony eksperyment

- Zbiór A: 10^5 losowych punktów o współrzędnych z przedziału $[-1000, 1000]$
- Zbiór B: 10^5 losowych punktów o współrzędnych z przedziału $[-10^{14}, 10^{14}]$
- Zbiór C: 1000 losowych punktów leżących na okręgu o środku (0,0) i promieniu $R=100$
- Zbiór D: 1000 losowych punktów o współrzędnych z przedziału $[-1000, 1000]$ leżących na prostej wyznaczonej przez wektor (a, b), $a = [-1.0, 0.0]$, $b = [1.0, 0.1]$

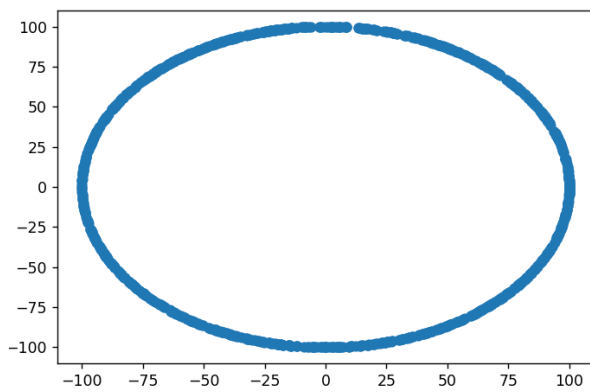
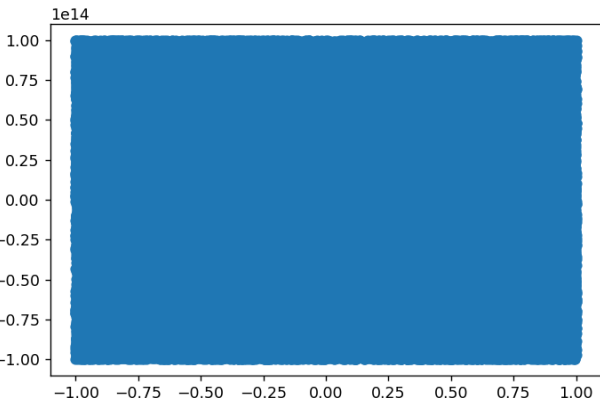
Współrzędne punktów zostały wygenerowane za pomocą funkcji `uniform()` z biblioteki `random`.

Wizualizacje zbiorów:

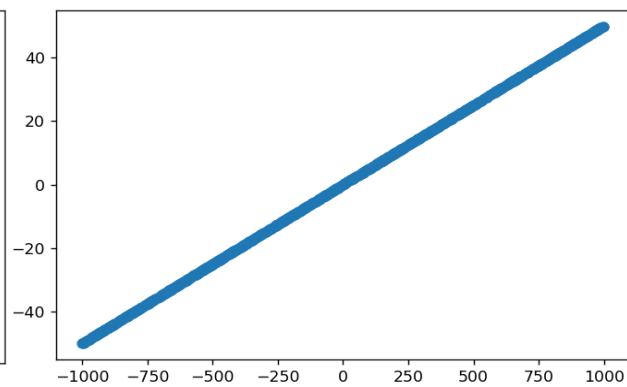
Rysunek 1: Zbiór A



Rysunek 2: Zbiór B



Rysunek 3: Zbiór C



Rysunek 4 Zbiór D

5. Wyniki eksperymentu

- Zbiór A

Dla zbioru A działania różnych metod wyznacznikowych zostały przetestowane dla następujących precyzji: $1e-10$, $1e-11$, $1e-12$, $1e-13$, $1e-14$. Wyniki klasyfikacji punktów względem prostej umieszczone zostały w tabeli poniżej.

Tabela 1: Wyniki klasyfikacji punktów ze zbioru A

	matrix_type	det_type	prec	Left	Right	Colinear
0	matrix_3x3	det_3x3	1.000000e-10	50042	49958	0
1	matrix_3x3	det from np	1.000000e-10	50042	49958	0
2	matrix_2x2	det_2x2	1.000000e-10	50042	49958	0
3	matrix_2x2	det from np	1.000000e-10	50042	49958	0
4	matrix_3x3	det_3x3	1.000000e-11	50042	49958	0
5	matrix_3x3	det from np	1.000000e-11	50042	49958	0
6	matrix_2x2	det_2x2	1.000000e-11	50042	49958	0
7	matrix_2x2	det from np	1.000000e-11	50042	49958	0
8	matrix_3x3	det_3x3	1.000000e-12	50042	49958	0
9	matrix_3x3	det from np	1.000000e-12	50042	49958	0
10	matrix_2x2	det_2x2	1.000000e-12	50042	49958	0
11	matrix_2x2	det from np	1.000000e-12	50042	49958	0
12	matrix_3x3	det_3x3	1.000000e-13	50042	49958	0
13	matrix_3x3	det from np	1.000000e-13	50042	49958	0
14	matrix_2x2	det_2x2	1.000000e-13	50042	49958	0
15	matrix_2x2	det from np	1.000000e-13	50042	49958	0
16	matrix_3x3	det_3x3	1.000000e-14	50042	49958	0
17	matrix_3x3	det from np	1.000000e-14	50042	49958	0
18	matrix_2x2	det_2x2	1.000000e-14	50042	49958	0
19	matrix_2x2	det from np	1.000000e-14	50042	49958	0

Podliczenie punktów które zostały sklasyfikowane dla tej samej precyzji obliczeń do dwóch różnych zbiorów wykazało, że niezależnie od wybranej metody liczenia wyznacznika (samodzielnie zaimplementowana funkcja, funkcja biblioteczna) , oraz typu wybranej macierzy (2x2,3x3) wszystkie punkty zostały sklasyfikowane w ten sam sposób, ponadto żaden z wygenerowanych punktów nie okazał się być współliniowym z podaną prostą, wynika to z faktu, że losowo wybrany punkt z płaszczyzny ma małe szanse znalezienia się dokładnie na konkretnej prostej.

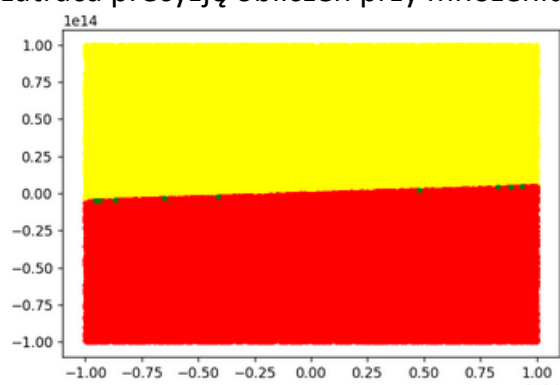
- Zbiór B

Dla tego zbioru testy zostały przeprowadzone na tych samych precyzjach co dla zbioru A. Wyniki klasyfikacji punktów w zbiorze przedstawia tabela poniżej.

Tabela 2 Wyniki klasyfikacji punktów ze zbioru B

	matrix_type	det_type	prec	Left	Right	Colinear
0	matrix_3x3	det_3x3	1.000000e-10	49711	50289	0
1	matrix_3x3	det from np	1.000000e-10	49711	50289	0
2	matrix_2x2	det_2x2	1.000000e-10	49705	50286	9
3	matrix_2x2	det from np	1.000000e-10	49711	50289	0
4	matrix_3x3	det_3x3	1.000000e-11	49711	50289	0
5	matrix_3x3	det from np	1.000000e-11	49711	50289	0
6	matrix_2x2	det_2x2	1.000000e-11	49705	50286	9
7	matrix_2x2	det from np	1.000000e-11	49711	50289	0
8	matrix_3x3	det_3x3	1.000000e-12	49711	50289	0
9	matrix_3x3	det from np	1.000000e-12	49711	50289	0
10	matrix_2x2	det_2x2	1.000000e-12	49705	50286	9
11	matrix_2x2	det from np	1.000000e-12	49711	50289	0
12	matrix_3x3	det_3x3	1.000000e-13	49711	50289	0
13	matrix_3x3	det from np	1.000000e-13	49711	50289	0
14	matrix_2x2	det_2x2	1.000000e-13	49705	50286	9
15	matrix_2x2	det from np	1.000000e-13	49711	50289	0
16	matrix_3x3	det_3x3	1.000000e-14	49711	50289	0
17	matrix_3x3	det from np	1.000000e-14	49711	50289	0
18	matrix_2x2	det_2x2	1.000000e-14	49705	50286	9
19	matrix_2x2	det from np	1.000000e-14	49711	50289	0

W tym zbiorze zachodzi anomalia dla wyznacznika 2x2 obliczanego funkcją zaimplementowaną samodzielnie, funkcja ta klasyfikuje 9 punktów (tych samych dla różnych precyzji) jako współliniowe z prostą. Po przeanalizowaniu tych punktów okazuje się posiadają one współrzędne z górnego zakresu generowania co może skłaniać nas do wniosku, że wpływ na tą anomalię ma arytmetyka pythona która ztraca precyzję obliczeń przy mnożeniu bardzo dużych liczb.



Rysunek 1 Anomalia Wyznacznika 2x2 dla zbioru B

- Zbiór C

Dla zbioru C otrzymane rezultaty są podobne jak dla zbioru A, żadne punkty nie zostały sklasyfikowane jako współliniowe i nie wystąpiły żadne anomalie. Niezależnie od obranej metody i precyzji obliczeń rezultaty są takie same. Wyniki klasyfikacji przedstawione w tabeli poniżej.

Tabela 3 Wyniki klasyfikacji punktów ze zbioru C

	matrix_type	det_type	prec	Left	Right	Colinear
0	matrix_3x3	det_3x3	1.000000e-10	499	501	0
1	matrix_3x3	det from np	1.000000e-10	499	501	0
2	matrix_2x2	det_2x2	1.000000e-10	499	501	0
3	matrix_2x2	det from np	1.000000e-10	499	501	0
4	matrix_3x3	det_3x3	1.000000e-11	499	501	0
5	matrix_3x3	det from np	1.000000e-11	499	501	0
6	matrix_2x2	det_2x2	1.000000e-11	499	501	0
7	matrix_2x2	det from np	1.000000e-11	499	501	0
8	matrix_3x3	det_3x3	1.000000e-12	499	501	0
9	matrix_3x3	det from np	1.000000e-12	499	501	0
10	matrix_2x2	det_2x2	1.000000e-12	499	501	0
11	matrix_2x2	det from np	1.000000e-12	499	501	0
12	matrix_3x3	det_3x3	1.000000e-13	499	501	0
13	matrix_3x3	det from np	1.000000e-13	499	501	0
14	matrix_2x2	det_2x2	1.000000e-13	499	501	0
15	matrix_2x2	det from np	1.000000e-13	499	501	0
16	matrix_3x3	det_3x3	1.000000e-14	499	501	0
17	matrix_3x3	det from np	1.000000e-14	499	501	0
18	matrix_2x2	det_2x2	1.000000e-14	499	501	0
19	matrix_2x2	det from np	1.000000e-14	499	501	0

Wnioski są analogiczne jak dla zbioru A.

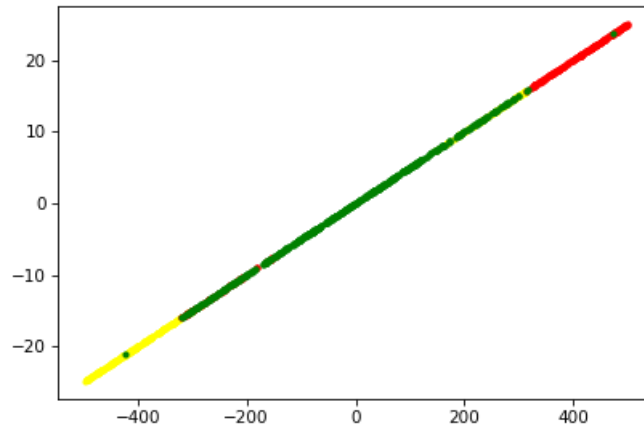
- Zbiór D

Punkty wygenerowane w tym zbiorze powinny być współliniowe z prostą na której zostały wygenerowane, analiza czy tak jest odbywa się na szerszym zakresie precyzji niż w poprzednich zbiorach a mianowicie na precyzjach: 1e-10,1e-11,1e-12,1e-13,1e-14,1e-15,1e-16,1e-17,1e-18. Wyniki otrzymanej klasyfikacji w tabeli poniżej.

Tabela 4 Wyniki klasyfikacji punktów ze zbioru D

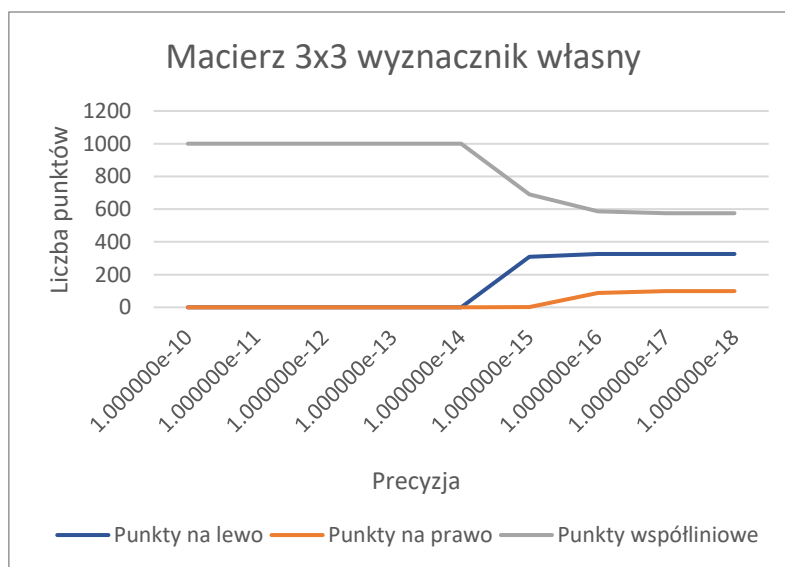
	matrix_type	det_type	prec	Left	Right	Collinear
0	matrix_3x3	det_3x3	1.000000e-10	0	0	1000
1	matrix_3x3	det from np	1.000000e-10	0	0	1000
2	matrix_2x2	det_2x2	1.000000e-10	0	0	1000
3	matrix_2x2	det from np	1.000000e-10	0	0	1000
4	matrix_3x3	det_3x3	1.000000e-11	0	0	1000
5	matrix_3x3	det from np	1.000000e-11	0	0	1000
6	matrix_2x2	det_2x2	1.000000e-11	0	0	1000
7	matrix_2x2	det from np	1.000000e-11	0	0	1000
8	matrix_3x3	det_3x3	1.000000e-12	0	0	1000
9	matrix_3x3	det from np	1.000000e-12	0	0	1000
10	matrix_2x2	det_2x2	1.000000e-12	41	31	928
11	matrix_2x2	det from np	1.000000e-12	14	18	968
12	matrix_3x3	det_3x3	1.000000e-13	0	0	1000
13	matrix_3x3	det from np	1.000000e-13	0	0	1000
14	matrix_2x2	det_2x2	1.000000e-13	149	121	730
15	matrix_2x2	det from np	1.000000e-13	270	247	483
16	matrix_3x3	det_3x3	1.000000e-14	0	0	1000
17	matrix_3x3	det from np	1.000000e-14	3	0	997
18	matrix_2x2	det_2x2	1.000000e-14	176	143	681
19	matrix_2x2	det from np	1.000000e-14	421	389	190
20	matrix_3x3	det_3x3	1.000000e-15	308	1	691
21	matrix_3x3	det from np	1.000000e-15	402	189	409
22	matrix_2x2	det_2x2	1.000000e-15	184	151	665
23	matrix_2x2	det from np	1.000000e-15	481	460	59
24	matrix_3x3	det_3x3	1.000000e-16	326	87	587
25	matrix_3x3	det from np	1.000000e-16	564	363	73
26	matrix_2x2	det_2x2	1.000000e-16	191	152	657
27	matrix_2x2	det from np	1.000000e-16	503	478	19
28	matrix_3x3	det_3x3	1.000000e-17	326	99	575
29	matrix_3x3	det from np	1.000000e-17	593	389	18
30	matrix_2x2	det_2x2	1.000000e-17	191	153	656
31	matrix_2x2	det from np	1.000000e-17	509	485	6
32	matrix_3x3	det_3x3	1.000000e-18	326	99	575
33	matrix_3x3	det from np	1.000000e-18	598	398	4
34	matrix_2x2	det_2x2	1.000000e-18	193	154	653
35	matrix_2x2	det from np	1.000000e-18	509	488	3

Analiza klasyfikacji punktów z zbioru D względem prostej, jest najbardziej interesująca, ponieważ wszystkie punkty w tym zbiorze zostały wygenerowane w ten sposób by znajdować się na tej prostej, jednak jak się okazuje w zależności od metody klasyfikowania punktów i przyjętej precyzji nie zawsze tak się działo.



Rysunek 2 Wynik klasyfikacji macierzą 2x2 przy precyzji $E=1e-14$
 Kolor żółty – punkty na lewo od prostej
 Kolor zielony – punkty na lewo od współliniowych
 Kolor czerwony – punkty na prawo od prostej

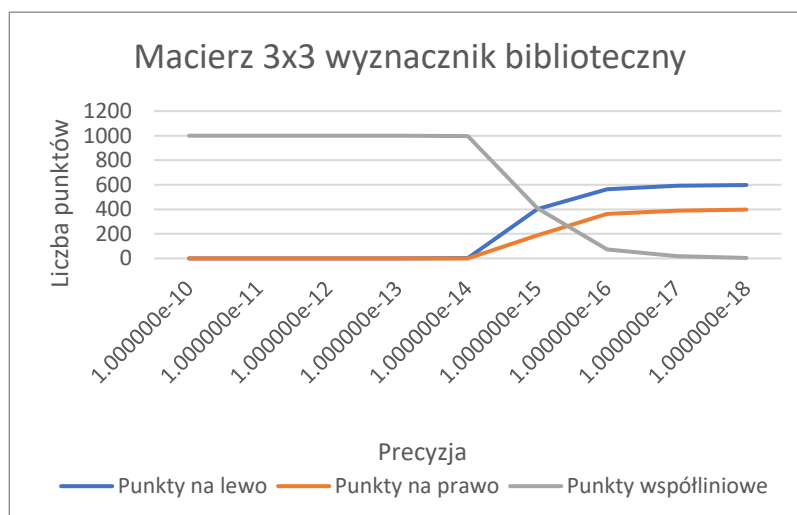
Pierwszą obserwacją jaka nasuwa się przy analizie wyników jest fakt, że metoda wykorzystująca macierz 2x2 i samodzielnie zaimplementowaną funkcję obliczającą wyznacznik, już przy precyzji $1e-12$ popełnia pierwsze błędy w klasyfikacji punktów na prostej. Jednak do wyciągania wniosków potrzebna jest bardziej pogłębiona analiza.



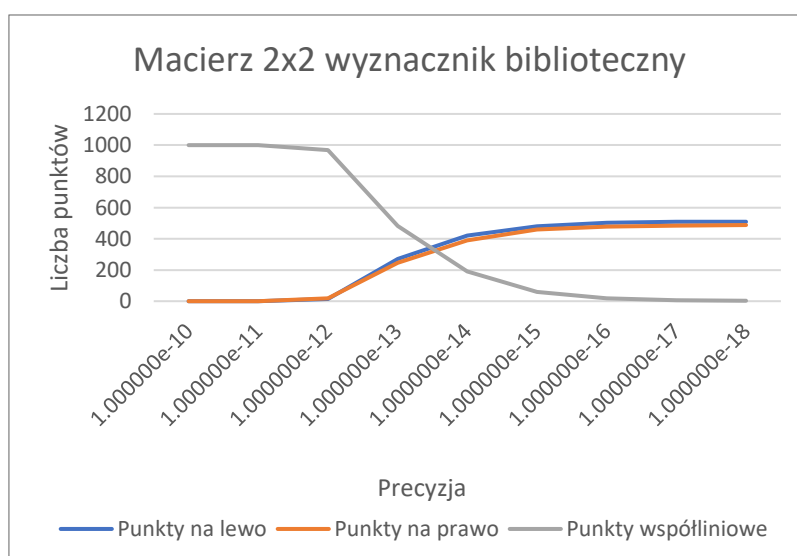
Wykres 1 Skuteczność metody z macierzą 3x3 i samodzielnie zaimplementowanym wyznacznikiem



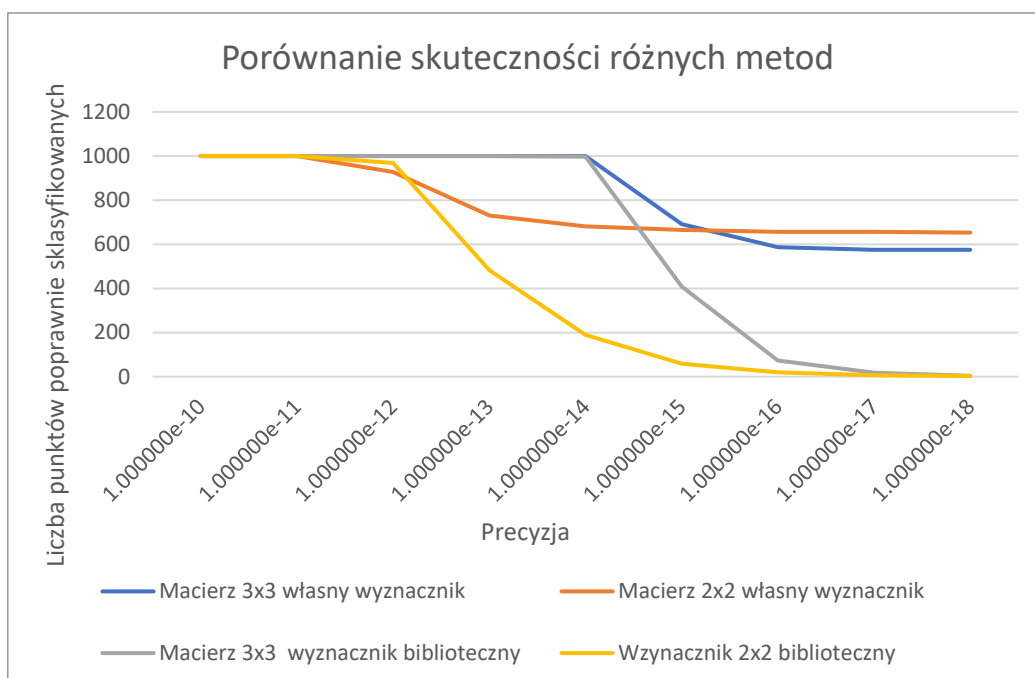
Wykres 2 Skuteczność metody z macierzą 2x2 i samodzielnie zaimplementowanym wyznacznikiem



Wykres 3 Skuteczność metody z macierzą 3x3 i wyznacznikiem z biblioteki numPy



Wykres 4 Skuteczność metody z macierzą 3x3 i wyznacznikiem z biblioteki numPy



Wykres 5 Porównanie skuteczności wszystkich metod

Wnioski

Na podstawie otrzymanych wyników można zauważyć, że w przypadku metod wykorzystujących macierz 3x3 obliczających wyznacznik zarówno funkcją zaimplementowaną samodzielnie jak i funkcją biblioteczną klasyfikacja punktów „najdłużej” odbywa się w sposób poprawny więc te właśnie metody możemy uznać za najskuteczniejsze. Warto również zauważyć, że funkcja zaimplementowana samodzielnie działa nieznacznie szybciej od funkcji bibliotecznej oraz tempo spadku jej skuteczności jest mniejsze.

6. Wnioski z doświadczenia

Na podstawie przeprowadzanego doświadczenia nasuwają się następujące wnioski:

- Wśród losowych punktów na płaszczyźnie istnieją małe szanse na to że punkty pojawiają się na konkretnej prostej
- Błędna arytmetyka zmiennie-przecinkowa komputera może mieć znaczący wpływ na wyniki obliczeń
- Dla bardzo dużych liczb arytmetyka komputera czasem zawodzi przy niektórych metodach klasyfikacji punktów
- Obrana precyzja obliczeń jest istotna dla wyniku
- W zależności od obranej metody segregacji punktów, możemy otrzymywać różne wyniki dla tych samych zbiorów. Więc wybór konkretnej metody powinien być wcześniej przemyślany

- Używanie funkcji bibliotecznych, kiedy nie jest to konieczne może nieznacznie spowolnić działanie programu. (Czasem lepiej samemu zaimplementować proste funkcje)
- Metoda klasyfikacji punktów wykorzystująca macierz 3x3 i samodzielnie zaimplementowaną funkcję obliczającą wyznacznik, w większości przypadków jest najlepszym wyborem.

Załączniki:

-Notebook Pythonowy z rozwiązaniami